

Introducció

El món dels videojocs està lligat al segle XXI. El sector dels videojocs va començar als anys 70 i es podria afirmar que es va començar a identificar com a tal l'any 1971, quan Nolan Bushnell y Ted Dabney van fundar Atari Computers i van llançar al mercat el primer videojocs oficial i comercial. A partir d'aquest moment, la indústria es va començar a desenvolupar i les màquines recreatives, i més tard, les consoles domèstiques i portàtils, varen començar a aparèixer i establint-se en el mercat.

El món dels videojocs és un dels sectors en la indústria que més capital general en l'àmbit de l'oci, amb uns ingressos que superen als generats per al cinema i la música junts.

Motivacions

La meua motivació personal per a realitzar aquest projecte ha estat la de conèixer i experimentar el desenvolupament dels videojocs des de dins, ja que és en l'àmbit dels videojocs a on vull enfocar la meua vida professional i realitzar aquest projecte pot ser una bona entrada per a aquest sector laboral.

Un dels meus objectius es poder arribar a publicar el projecte una vegada finalitzat. Gràcies als *markets* mòbils i al seu model de negoci és més fàcil distribuir, encara que es tinguin pocs recursos. Per PC també existeix quelcom semblant, *Steam Greenlight*, una plataforma a on es poden penjar els projectes *indie* i, amb els suficients vots de la comunitat, aquest pot sortir a la venda a la plataforma *Steam*.

Metodologia

Scrum és una metodologia de desenvolupament àgil, que està basat en el desenvolupament iteratiu i incremental. Durant cada *sprint* (període d'entre una i quatre setmanes), l'equip crea un increment de software "potencialment entregable". Aquestes característiques són un subconjunt del *Product Backlog*, que és un conjunt de requisits d'alt nivell prioritzats que defineixen la feina a fer. Els elements del *Product Backlog* que formen part de l'*sprint* es determinen a la reunió d'*Sprint Planning*.

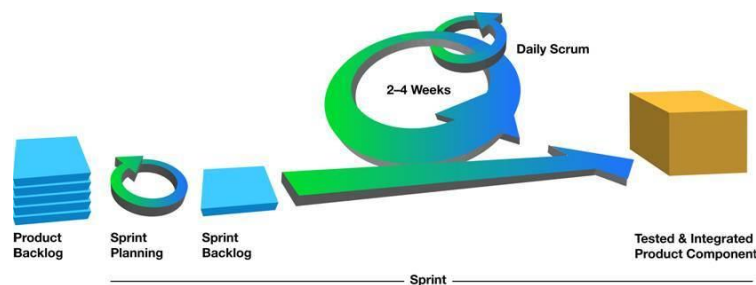


Figura 1: Marc de treball SCRUM

Beneficis de l'Scrum

- **Flexibilitat contínua als canvis:** Gran capacitat de reacció davant els requeriments canviants generats per les necessitats del client o l'evolució del mercat.
- **Reducció del Time to Market:** El client pot començar a utilitzar les característiques més importants del projecte abans de que estigui completament acabat.
- **Millor qualitat del software:** El treball metòdic i la necessitat d'obtenir una versió del treball funcional després de cada iteració ajuda a l'obtenció d'un software d'alta qualitat.
- **Major productivitat:** S'aconsegueix, en altres raons, degut a l'eliminació de la burocràcia i la motivació de l'equip proporcionada pel fet de que es poden estructurar de manera autònoma.

- **Predicció dels temps:** A través d'aquest marc de treball es coneix la velocitat mitjana de l'equip per sprint, amb els que es possible estimar de manera fàcil quan es podrà fer ús d'una determinada funcionalitat que encara es troba en el Backlog.
- **Reducció de riscos:** El fet de dur a terme les funcionalitats de més valor al principi i saber la velocitat en la que avança l'equip permet detectar riscos innecessaris de manera anticipada.

La programació de videojocs

A la figura 2 podem observar un esquema de com funciona l'execució d'un videojoc. El seguiment de l'execució comença amb un mètode **Start** que inicialitza totes les dades i eines que s'utilitzin (variables, elements, connectors amb la targeta gràfica si es necessitessin, etc). A cada tic de rellotge, el sistema llegirà els inputs del jugador, actualitzarà l'estat del joc (col·lisions, intel·ligència artificial, física, etc) i pintarà el resultat per pantalla. En cas de que el joc provoqui un game over, el sistema s'encarregarà de guardar les puntuacions, tancar els fitxers que s'hagin obert i de finalitzar el que sigui necessari.

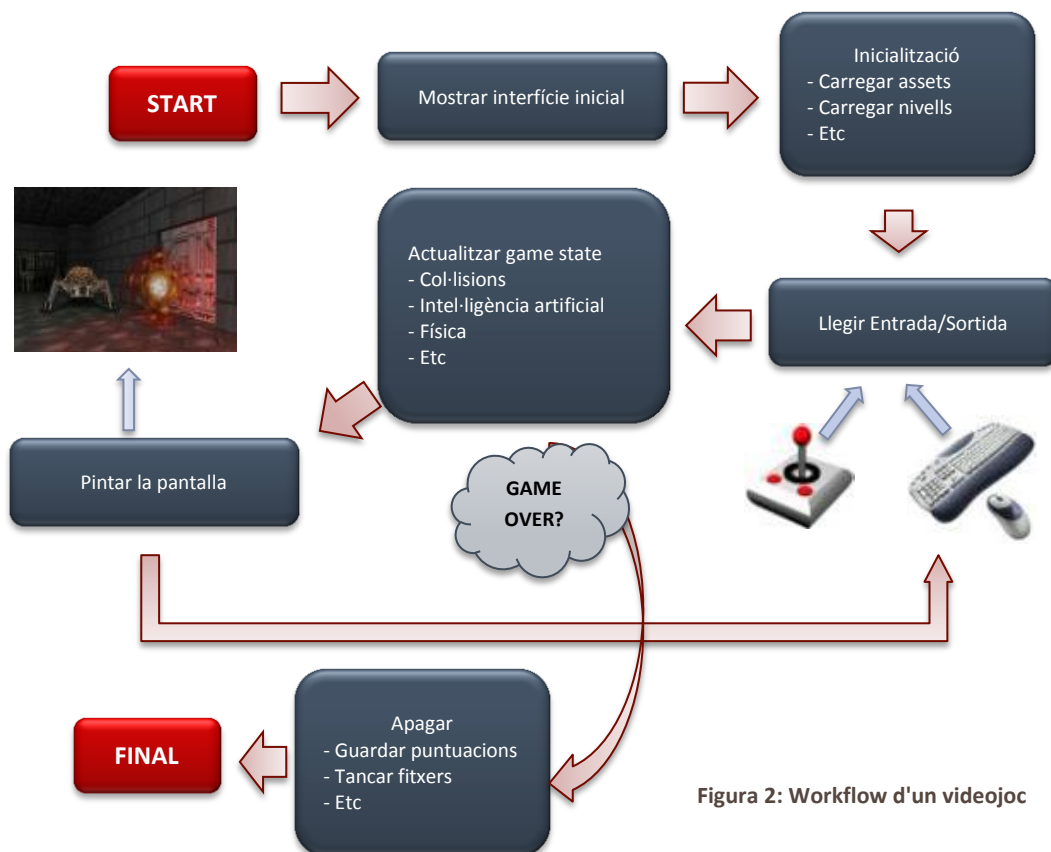


Figura 2: Workflow d'un videojoc

Unity3D



Unity3D és una eina integrada d'edició 3D per a crear tan videojocs o altres continguts interactius, com per exemple, visualitzacions arquitectòniques o animacions 3D en temps real. En el sentit de que

l'entorn gràfic integrat és el mètode principal de desenvolupament, és similar a altres programes com el Director o el Blender game engine. Tot i que el Unity3D té l'entorn gràfic integrat, també té integrades la física i altres elements per fer que, més que un motor gràfic, sigui un motor de videojocs. Com a motor gràfic, és possiblement el millor que està per sota dels 10.000€.

S'ha escollit Unity3D perquè, a part de l'experiència prèvia amb el motor, amb la nova versió 4.3 permet realitzar jocs directament en dues dimensions, que ocasionen que el motor "ignori" la tercera dimensió i faciliti la implementació del joc. Un altre motiu és que permet una visualització instantània de l'escena i que, gràcies a la "Assets Store" podem trobar multitud d'exemples, tutorials i eines per a facilitar el desenvolupament. Sobre l'elecció del llenguatge s'ha escollit C# davant de Javascript per la millor documentació que es pot trobar a la xarxa i la facilitat de programació amb les funcions d'autocompletar de l'editor de Unity, a més de la familiaritat amb el llenguatge. Boo s'ha descartat des del principi per tractar-se d'un llenguatge inferior i, per tant, que té molta menys documentació que els altres dos.

Anàlisi i disseny

A la figura 4 hi ha un diagrama de cas d'ús que mostra totes les opcions que té el jugador a la partida: saltar, colpejar, pausa el joc. A més de les funcions que realitza el sistema, que són la detecció de col·lisions i moure el personatge automàticament cap endavant.

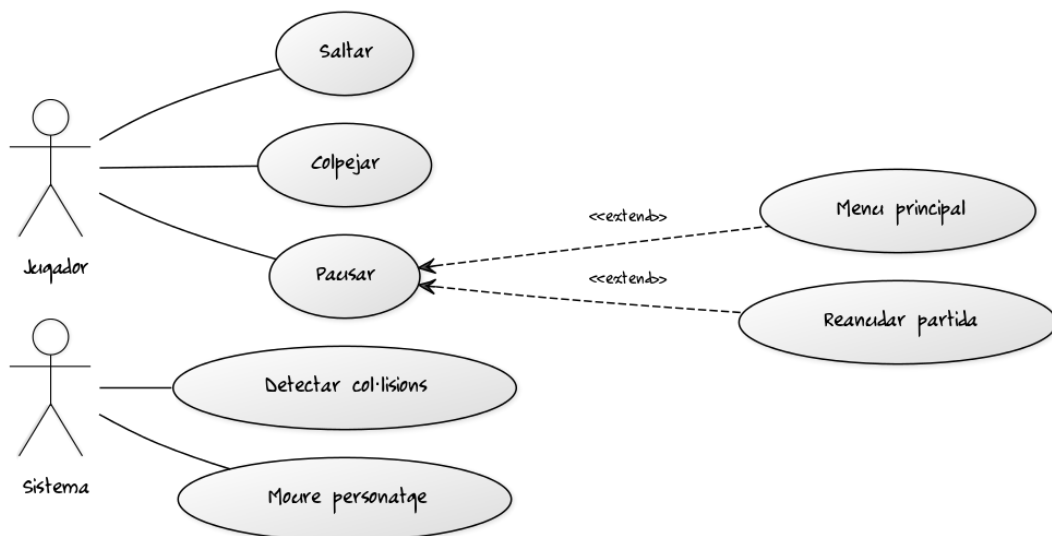


Figura 4: Diagrama de cas d'ús de joc

A la figura 5 es pot observar el diagrama d'activitat que segueix el sistema en la detecció i resposta de les col·lisions. Les possibilitats són: un objecte destruïble, una llum, el terra o una moneda. El sistema respondrà d'una manera o d'una altra segons l'objecte col·lisionat.

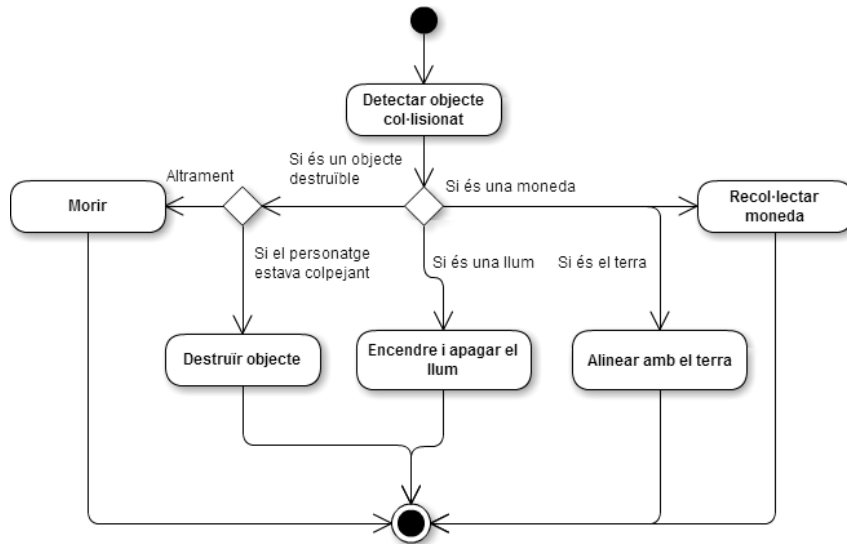


Figura 5: Diagrama d'activitat del càlcul de col·lisions

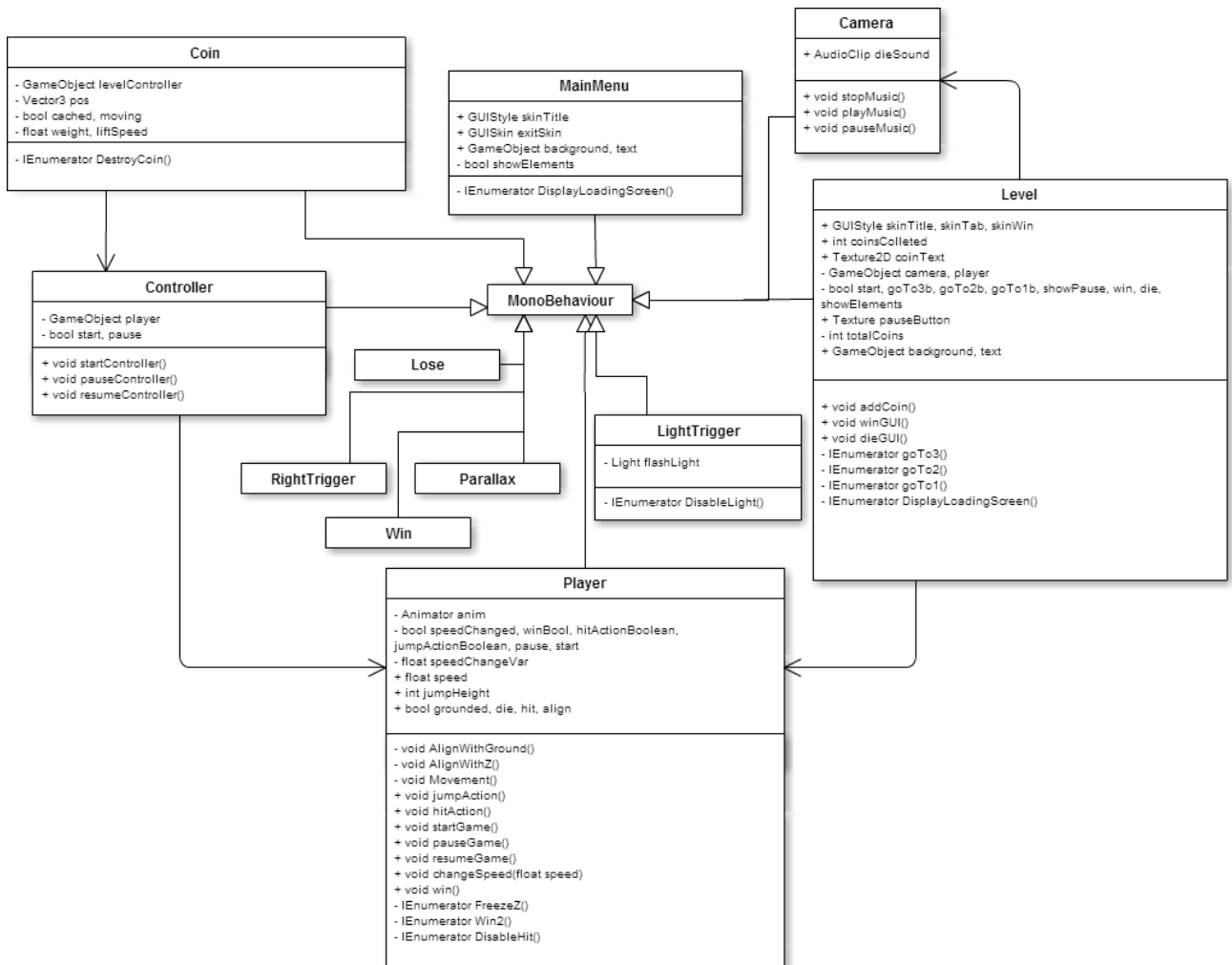


Figura 2: Diagrama de classes de l'aplicació



Conclusions

Durant l'elaboració d'aquest Projecte Final de Carrera s'han arribat a varies conclusions:

- És molt important usar les eines adequades i realitzar un bon anàlisi dels problemes que poden sorgir. Tot i que suposi una inversió de temps extra, es nota en el resultat final i ajuda a solucionar molts problemes.
- Realitzar un videojoc no és només modelar i picar codi. Possiblement el moment més important en la creació d'un videojoc és la fase de dissenyar els elements i les regles que el regeixen, o sigui, realitzar un bon GDD (Game Document Design). Aquesta feina en un desenvolupament professional la porta a terme un Game Designer.
- Una persona sola no pot desenvolupar un gran projecte. En el desenvolupament d'un videojoc hi intervenen persones amb diferents rols, com per exemple, el dissenyador gràfic i el compositor.